Comments: in case of all programs assume that libraries iostrem and stdlib are attached and the namespace std is available. Only places marked for the answers are graded. In case you find a mistake or ambiguity in a question, write an appropriate comment which explains that. The number of marks fo the test is 0-100 points (passing threshold = 50%).

Question 1. (20 pts. = 2*10 pts.)

Fill out the selected fields so that the depicted calls to the function *f* return the given values:

f(2) = 1

f(4) = 3

f(9) = 8

Note: in case of multiple correct answers give the one with minimum number of characters (shortest one).

```
int f( int x ) {

    if (        x>0          )

            return x + f(    -1     );
        else
            return x;
}
```

Question 2. (21 pts. = 3*7 pts.)

Give the text that is written to the screen as a result of executing the subsequent instructions *cout* (in place for an answer marked with label "Instruction x:" write the text that is printed to the screen by instruction *cout* makred with comment /* I-x */. Write *ERR* if the answer cannot be uniquely determined.

Answers:
  Instruction 1: **0**

  Instruction 2: **4**

  Instruction 3: **0**

```
typedef struct {
    int x, y, *ptr;
} type_t;
void f( type_t *p ) {
    p->ptr = &(p->x);
    p->y = 4;  }
void g( type_t p ) {
    p.ptr = &(p.x);
    p.x = p.y = 5;  }

type_t p;
int main() {
  g( p );
  f( &p );
  cout << p.x; /* I-1 */
  cout << p.y; /* I-2 */
  cout << *(p.ptr); /* I-3 */
  return 0;
}
```

Question 3. (20 pts. = 4*5 pts.)

Fill out the selected fields so that the function returns the value of the field *hash* in the last structure in the table *b* whose field *key* equals *x*. The function returns -1 if the table *b* has no structure whose field *key* equals *x*.

Note: in case of multiple correct answers give the one with minimum number of characters (shortest one). Write in all fields *ERR* if the program cannot do what instructed.

```
typedef struct {
    int key;
    int hash;
} pairs_t;

int find( pairs_t      ERR    , int n, int x ) {

    while (      ERR        >= 0 )

        if ( b     ERR        key ==       ERR        )
             return n;
    return -1;
}
```

Question 4. (20 pts. = 5*4 pts.)
Give the text that is written to the screen as a result of
executing the subsequent instructions *cout* (in place for an
answer marked with label "Instruction x:" write the text
that is printed to the screen by instruction *cout* makred
with comment /* I-x */. Write *ERR* if the answer cannot be
uniquely determined. Binary encoding of numbers is
assumed, as presented during the lectures, i.e., U2. If
some instruction results in an execution error, then also
write *ERR* as an answer and continue your analysis by
skipping this instruction.

```
int main() {
 int t[] = {1,2,3,4};
 int a=5, *x = &(t[2]);

 cout << (a<<2) + (a+1); /* I-1 */
 if ( *t - *(t+1) || a-- )
     a--;
 cout << a; /* I-2 */
 cout << (8^6); /* I-3 */
 cout << *t + *(x--); /* I-4 */
 cout << sizeof x; /* I-5 */
 return 0;
}
```

Answers:

Instruction 1:   26

Instruction 2:   4

Instruction 3:   14

Instruction 4:   4

Instruction 5:   ERR

Question 5. (19 pts.)
What the following program will write
to the screen? Write *ERR* if the answer
cannot be uniquely determined, or the
program has compilation or execution
errors.

Answer: **LkLL**

```
char s[128] = { "WeLikeLanguageC" };
int move( char *t, int i ) {
    int j;
    for ( j=i; j < i+5; j++ )
        if ( t[j] == '\0' )
            return i;
    return i + 3 - (i%2);
}

int main() {
    int i=0, k;

    do {
        k = i;
        i = move( s+i, i );
        cout << *(s+i-1);
    } while ( i != k );
    return 0;
}
```